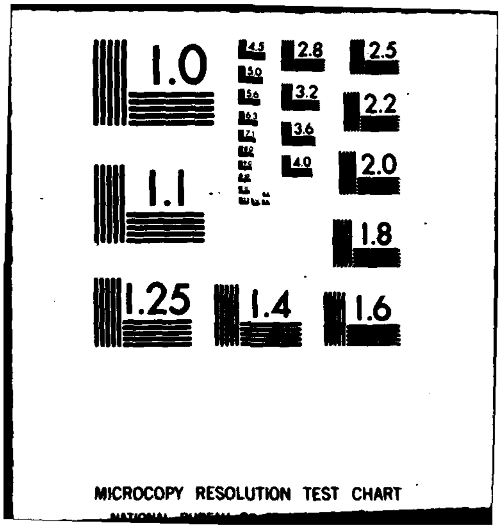
AD-A093 732   STANFORD UNIV CA   F/G 9/2
COMPUTATIONAL COMPLEXITY, EFFICIENCY AND ACCOUNTABILITY IN LARG--ETC(U)
DEC 80 J T GILL, M E HELLMAN   DAAG29-78-C-0036
UNCLASSIFIED   ARO-15843.3-EL   NL

1 of 1
AD A093732

END
DATE
FILMED
2-81
DTIC


1.0
4.5
5.0
5.6
6.3
7.1
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6
MICROCOPY RESOLUTION TEST CHART

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

LEVEL II

12

AD A093732

REPORT DOCUMENTATION PAGE	**READ INSTRUCTIONS BEFORE COMPLETING FORM**
1. REPORT NUMBER 19 15843.3-EL	2. GOVT ACCESSION NO. AD-A093732
3. RECIPIENT'S CATALOG NUMBER 9	
4. TITLE (and Subtitle) 6 Computational Complexity, Efficiency and Accountability in Large Scale Teleprocessing Systems.	5. TYPE OF REPORT & PERIOD COVERED Final Report. 1 Sep 78 — 31 Oct 80 16
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) 10 John T./Gill Martin E./Hellman	8. CONTRACT OR GRANT NUMBER(s) 15 DAAG29-78-C-0036 new ok
9. PERFORMING ORGANIZATION NAME AND ADDRESS Stanford University Stanford, CA 94305	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS U. S. Army Research Office Post Office Box 12211 Research Triangle Park, NC 27709	12. REPORT DATE 11 Dec 80
	13. NUMBER OF PAGES 4
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) 18 ARO 12 16	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

DTIC SELECTE D JAN 13 1981 S E

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

NA

18. SUPPLEMENTARY NOTES

The view, opinions, and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

computation
teleprocessing systems
digital signals
random graphs
factoring

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

→ Research is summarized in the following areas: certified digital signals, factoring and random graphs, compact knapsacks, NP-complete problems, and indices in a finite field GF(q^m). ←

q to the m power

DDC FILE COPY

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

332550 Jlu

81 1 12 051

FINAL REPORT

COMPUTATIONAL COMPLEXITY, EFFICIENCY AND ACCOUNTABILITY IN LARGE SCALE TELEPROCESSING SYSTEMS

DAAG29-78-C-0036

STANFORD UNIVERSITY

JOHN T. GILL

MARTIN E. HELLMAN

ARO PROPOSAL NUMBER: P-15843-EL

1. Research Progress

A. Certified Digital Signatures

Digital signatures provide a logical equivalent to written signatures yet can be transmitted over normal electronic communication channels, e.g., telephones, radios, digital networks, etc. They hold great promise for electronic communications by allowing new levels of authentication and accountability in dealings conducted over telecommunication channels.

Although public key cryptosystems can be used to generate digital signatures, certification of public key cryptosystems is a major problem. We have developed a new digital signature system which is "pre-certified," in the sense that it only depends on the existence of a one-way function. (A conventional cryptosystem can even be used to generate a one-way function, and many such systems are available and already certified.) The new method generates signatures of about 15 kilobits (2 kilobytes), requires a few thousand applications of the underlying one-way (or encryption) function per signature, and only a few kilobytes of memory.

B. Factoring and Random Graphs

The problem of factoring large numbers has interested mathematicians since ancient times, and has gained additional practical importance through recently developed public key cryptosystems which depend on the difficulty of factoring for their security. The fastest factoring method known at present is due to Richard Schroeppel, and he has suggested an improvement which he thought might speed it up even further. We have analyzed his suggested improvement and shown that it should not increase the speed of the algorithm.

Schroeppel's factoring method depends on finding a set of binary n-vectors (vectors with n entries, each either 0 or 1) which are dependent (one of them can be written as a binary sum of the others). His improvement generates a set of such vectors with only two 1's and n-2 0's.

It is known that, if the vectors had a single 1 and n-1 0's, then only about square root of n vectors would be needed before a dependent subset could be found. (This is an instance of the "Birthday Problem". Only 23 people are needed in a room for there to be better than a 50% chance of at least two of them having a birthday in common. If there were n days in the year, about square root of n people would be needed before there would be appreciable chance of an overlap.)

If the vectors were chosen with half 0's and half 1's, then about n vectors would be needed before a dependence would be expected.

Because the vectors generated by Schroeppel's modification have two 1's, it might be hoped that the behavior would be close to that of vectors with a single 1, in which case only about square root of n of them would be needed before a dependence would be expected, and the modification would increase the speed of factoring. If, however, the behavior of vectors with two 1's were closer to that of vectors with half 1's and half 0's (where about n are needed for a dependence), the modification would not increase speed.

We have recast this question in terms of graph theory and have utilized a result of Erdos and Renyi on the evolution of random graphs to show that approximately n vectors are needed in the modification, so it is not an improvement. In setting up the equivalence, dependent sets of vectors are equated to complete cycles in the graphs. For example, a graph which connects node 2 to node 3, node 3 to node 5, and node 5 back to node 2 possesses a complete cycle.

In our equivalence, a connection between two nodes corresponds to choosing a vector with its two 1's in those locations. For example, the vector 01100 corresponds to connecting nodes 2 and 3, 00101 corresponds to connecting node 3 to node 5, and 01001 corresponds to connecting node 5 back to node 2. It is seen that the complete cycle (2 to 3 to 5 to 2) corresponds to a dependent subset since the first two vectors add to 01201, which is the same as the third vector 01001 in binary arithmetic.

The question of how many binary n-vectors are needed before finding a dependent subset becomes the question of how many edges are need in a random graph with n vertices before a complete cycle occurs. Erdos and Renyi showed that $O(n)$, not $O(n^{1/2})$ are needed. Vectors with two 1's therefore behave almost the same as vecotrs with n/2 1's so far as dependence is concerned.

C. Compact Knapsacks

We are also investigated fast methods for solving "compact knapsack" problems: Given S and an n-vector of integers, a, find an n-vector, x, with each component an integer in the range (0,B), such that a*x = S. The usual knapsack problem corresponds to B=1, in which case x is a binary n-vector. Compact knapsacks have an advantage in that a smaller a vector compresses a greater amount of data in a one-way fashion. Although our results are not complete, they indicate that the complexity of solving compact knapsacks grows exponentially in n but only polynomially in b, where B=2**b. This further indicates that compact knapsacks may gain little over binary knapsacks.

D. NP-complete Problems

In our study of NP-complete problems, we attacked a major unanswered question in the theory of computation: "Do there exist computational problems for which it is easy to check that a proposed solution is correct, but for which it is in general very difficult to find the correct solution?" Most investigators believe that the answer to this question is affirmative, but this has not yet been proven. A negative answer would imply that many problems thought to be difficult, such as factorization of large numbers and optimal scheduling or routing, would be easily solved.

We provided evidence that the answer to the above question is affirmative by studying computers which have access to a tape of random numbers. For such computers, and for almost all tapes of random numbers, we found problems that are hard to solve but easy to check. We have also suggested how such random tapes can be simulated by deterministically generating "pseudorandom" numbers by a complicated but efficient computer program, thus introducing a class of problems that may someday be shown to be hard to solve but easy to check.

E. Indices in GF(q^m)

The problem of computing indices in a finite field GF(q^m) is of importance to the Diffie-Hellman public key distribution system. Pohlig and Hellman developed an improved algorithm for this problem, but it is only of value for a small fraction of finite fields. Merkle and Adleman have developed another algorithm which is generally applicable in GF(q). *We have extended this algorithm to deal with extension fields GF(q^m).* Irreducible polynomials play the role of primes, and the subexponential computation time of the Merkle-Adleman algorithm is retained.

2. Publications

R. Merkle, "A certified digital signature," submitted to Communications of the ACM.

M. Hellman, "On the difficulty of computing logarithms over GF(q^m)," 1980 Symposium on Security and Privacy, Oakland, CA, April 14-16, 1980.

J. Reyneri and M. Hellman, "On factoring and random graphs," submitted to Communications of the ACM.

Students Supported

Ralph Merkle (Ph.D. awarded June 1979)

Justin Reyneri

David Gluss